MACHINE LEARNING SERIES

Clustering and Unsupervised Learning

Which ones are supervised learning?

- Decision trees
- Artificial neural networks
- K-Nearest Neighbor
- Support vectors
- Linear regression
- Logistic regression
- …

This is how you learn supervised

- F(x): true function (usually not known)
- D: training sample drawn from F(x)
- G(x): model learned from training sample D
- Goal: $E<(F(x) - G(x))^2>$ is small (near zero) for future samples drawn from F(x)

How do you learn unsupervised???

- You don't train anything.
- You have no Test set to verify your model.
- You have some data and then you do things to it.
- How do you learn???
- How do you know what the goal is???
- What criteria do you specify to know how well you're doing???
- OMGZPONIES (o_O)

What to Learn/Discover?

- Statistical Summaries
- Generators
- Density Estimators
- Patterns/Rules
- Associations
- Clusters/Groups
- Exceptions/Outliers
- Changes in Patterns over time or location

Goals and Performance Criteria?

- Statistical Summaries
- Generators
- Density Estimators
- Patterns/Rules
- Associations
- Clusters/Groups
- Exceptions/Outliers
- Changes in Patterns over time or location

Clustering

- Given:
  - Data set D (training set)
  - Similarity/distance metric/information
- Find:
  - Partitioning of data
  - Groups of similar/close items

Similarity?

- Groups of similar customers
  - Similar demographics
  - Similar buying behavior
  - Similar health
- Similar products
  - Similar cost
  - Similar function
  - Similar store
- Similarity usually is domain/problem specific

Types of Clustering

- Partitioning
  - K-means clustering
  - K-medoids clustering
  - EM (expectation maximization) clustering
- Hierachical
  - Divisive clustering (top-down)
  - Agglomerative clustering (bottom-up)
- Density-Based methods
  - Regions of dense points separated by sparser regions

Types of Clustering

- Hard clustering
  - Each object is in one and only one cluster
- Soft clustering
  - Each object has probability of being in each cluster

Agglomerative Clustering

- Put each item in its own cluster
- Find all pairwise distances between clusters
- Merge the two closest clusters
- Repeat until everything is in one cluster

- Hierarchical clustering
- Yields a clustering with each possible # of clusters
- Greedy clustering: not optimal for any cluster size

Merging: Closest Clusters

- Nearest centroids
- Nearest medoids
- Nearest neighbors (shortest link)
- Nearest average distance (average link)
- Smallest greatest distance (maximum link)
- Domain specific similarity measure
  - Word frequency, TFIDF, KL-divergence, …
- Merge clusters that optimize criterion after merge

Agglomerative Clustering

- Greedy clustering
  - Once points are merged, they are never separated
  - Suboptimal with respect to clustering function
- Combine greedy with iterative refinement
  - Post processing
  - Interleaved refinement

Agglomerative Clustering

- Computational Cost
  - $O(n^2)$ just to read/calculate pairwise distances
  - N-1 merges to build complete hierarchy
    - Scan pairwise distances between clusters
    - Calculate pairwise distances between clusters
    - Fewer clusters to scan as clusters get larger
  - Overall $O(n^3)$ for simple implementations
- Improvements
  - Sampling
  - Dynamic sampling: add new points while merging

K-Means Clustering

- Inputs: data set and k (number of clusters)
- Output: each point assigned to one of k clusters
- K-Means algorithm:
  - Initialize the k-means
    - Assign from randomly selected points
    - Randomly or equally distributed in space
  - Assign each point to nearest mean
  - Update means from assigned points
  - Repeat until convergence

K-Means Clustering: Convergence

- Squared-Error Criterion
- Converged when SE criterion stops changing
- Increasing K reduces SE – can't determine K by finding minimum SE
- Instead, plot SE as a function of K

K-Means Clustering

- Efficient
  - K << N, so assigning points is O(K*N) < O(N^2)
  - Updating means can be done during assignment
  - Usually number of iterations << N
  - Overall O(N*K*iterations) closer to O(N) than O(N^2)
- Gets stuck in local minima
  - Sensitive to initialization
- Number of clusters must be pre-specified
- Requires vector space data to calculate means

Soft K-Means Clustering

- Instance of EM (expectation maximization)
- Like K-Means, except each point is assigned to each cluster with a probability
- Cluster means updated using weighted average
- Generalizes to Standard_Deviation/Covariance
- Works well if if cluster models are known

Soft K-Means (EM)

- Initialize model parameters:
  - Means
  - Std_devs
  - …
- Assign points probabilistically to each cluster
- Update cluster parameters from weighted points
- Repeat until convergence to local minimum

Cost of K-Means Clustering

- n cases; d dimensions; k centers; i iterations
- Compute distance each point to each center: O(ndk)
- Assign each of n cases to closest center: O(nk)
- Update centers (mean) from assigned points: O(ndk)
- Repeat i times until convergence
- Overall: O(ndki)
- Much better than $O(n^2)$-$O(n^3)$ for HAC
- Sensitive to initialization – run this many times
- Usually don't know k – run many times with diff k
- Requires many passes through data set